

Hoja de ejercicios de Enumerados y Arrays

1. Declara un tipo enumerado `tCalificacion` con los valores `noPresentado`, `suspense`, `aprobado`, `notable`, `sobresaliente` y `matriculaDeHonor`.

Luego, declara dos variables `nota1` y `nota2`, y guarda en ellas dos calificaciones numéricas de 0 a 10 (con un decimal) introducidas por el usuario.

Asigna a dos variables `calif1` y `calif2`, de tipo `tCalificacion`, el valor que les corresponda, de acuerdo con los valores de las correspondientes variables numéricas que se han guardado en `nota1` y `nota2` (0: `noPresentado`).

Finalmente, muestra cada nota numérica seguida de la calificación textual que le corresponde. Sólo mostrará las calificaciones si ambas notas están entre 0 y 10.

El programa usará las funciones adecuadas.

Ejemplo de ejecución:

```
Nota 1: 0
Nota 2: 10
Nota 1: 0 <No presentado>
Nota 2: 10 <Matricula de Honor>
```

2. Implementa un programa que pida al usuario las calificaciones de los 10 estudiantes de una clase y los guarde en un array (declara el tipo adecuado; se admiten decimales). Después, mostrará esas calificaciones (cada una en una línea), seguida de la media del curso y del número de aprobados y suspensos. Se usará una función para calcular la nota media y otra para el número de suspensos.

Ejemplo de ejecución:

```

Introduce la nota del estudiante 1: 7.75
Introduce la nota del estudiante 2: 8.25
Introduce la nota del estudiante 3: 10
Introduce la nota del estudiante 4: 9
Introduce la nota del estudiante 5: 3.25
Introduce la nota del estudiante 6: 6.30
Introduce la nota del estudiante 7: 2.25
Introduce la nota del estudiante 8: 7
Introduce la nota del estudiante 9: 6.95
Introduce la nota del estudiante 10: 8.25
Nota del estudiante 1 : 7.75
Nota del estudiante 2 : 8.25
Nota del estudiante 3 : 10.00
Nota del estudiante 4 : 9.00
Nota del estudiante 5 : 3.25
Nota del estudiante 6 : 6.30
Nota del estudiante 7 : 2.25
Nota del estudiante 8 : 7.00
Nota del estudiante 9 : 6.95
Nota del estudiante 10: 8.25

La nota media es: 6.90

Hay 2 suspensos y 8 aprobados

```

3. Dado el siguiente tipo tVector para representar secuencias de N enteros:

```
const int N = 10;
```

```
typedef int tVector[N];
```

- Escribe una función que calcule y devuelva la suma de los elementos que se encuentran en las posiciones pares del array.
 - Escribe una función que dados dos tVector devuelve un valor que indica si son iguales.
 - Crea una función principal (main) donde pruebes todas las funciones creadas en los apartados anteriores.
4. Modifica el programa anterior para que se puedan guardar hasta 100 calificaciones. El programa pedirá calificaciones al usuario hasta que éste introduzca una nota negativa (o se llene el array).

Ejemplo de ejecución:

```

Introduce la nota del estudiante 1: 5.25
Introduce la nota del estudiante 2: 3
Introduce la nota del estudiante 3: 6.65
Introduce la nota del estudiante 4: 9
Introduce la nota del estudiante 5: 10
Introduce la nota del estudiante 6: 2.30
Introduce la nota del estudiante 7: -1
Nota del estudiante 1 : 5.25
Nota del estudiante 2 : 3.00
Nota del estudiante 3 : 6.65
Nota del estudiante 4 : 9.00
Nota del estudiante 5 : 10.00
Nota del estudiante 6 : 2.30

La nota media es: 6.03

Hay 2 suspensos y 4 aprobados

```

5. Escribe un programa en C++ que lea de un archivo palabras.txt (con xxx como centinela) una lista de hasta 50 palabras (cadenas) y luego muestre al usuario un menú con estas opciones:

- Mostrar la lista de palabras (una por línea precedida de su posición).

```
1 - Mostrar la lista
2 - Localizar una palabra
0 - Salir
opcion: 1

1 Hola
2 Madrid
3 una
4 Ello
5 Frankfurt
6 Abracadabra
7 una
8 Venta
9 Computadora
10 una
11 entre
```

- Localizar una palabra: el usuario indicará la posición de la palabra y se mostrará por pantalla la palabra que ocupa dicha posición.

```
1 - Mostrar la lista
2 - Localizar una palabra
0 - Salir
opcion: 2

Palabra a localizar: Madrid
Localizada en la posicion 2

1 - Mostrar la lista
2 - Localizar una palabra
0 - Salir
opcion: 2

Palabra a localizar: Segovia
No encontrada
```

- Salir.

El programa repetirá el menú hasta que se elija la opción para salir y usará las funciones adecuadas.

